

# Programmation LabVIEW 8.6.1 FIRST

2011-2012

Équipe : Évolution 2626



# FRC cRIO Robot Project

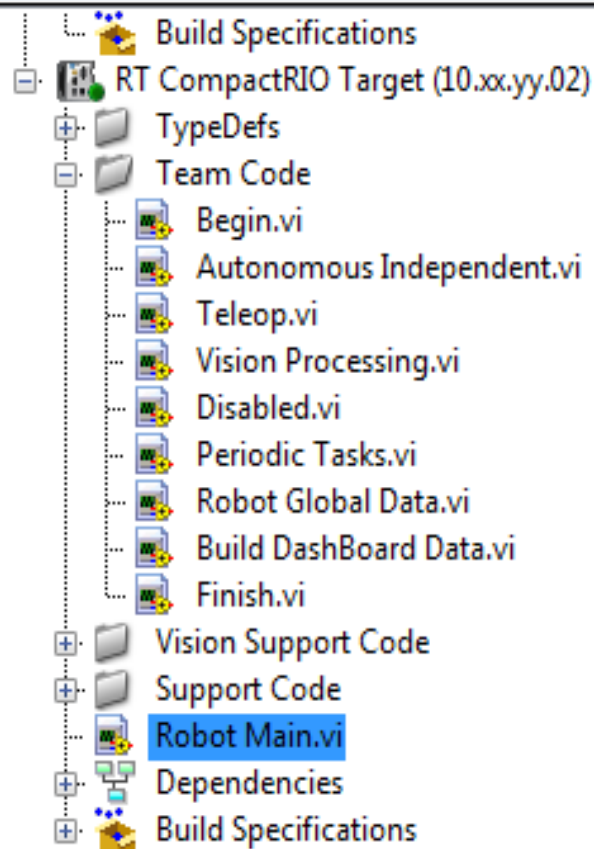
- Robot Main
- Begin
- Autonomous Independent
- Teleop
- Vision Processing
- Disabled
- Periodic task
- Finish

# FRC cRIO Robot Project

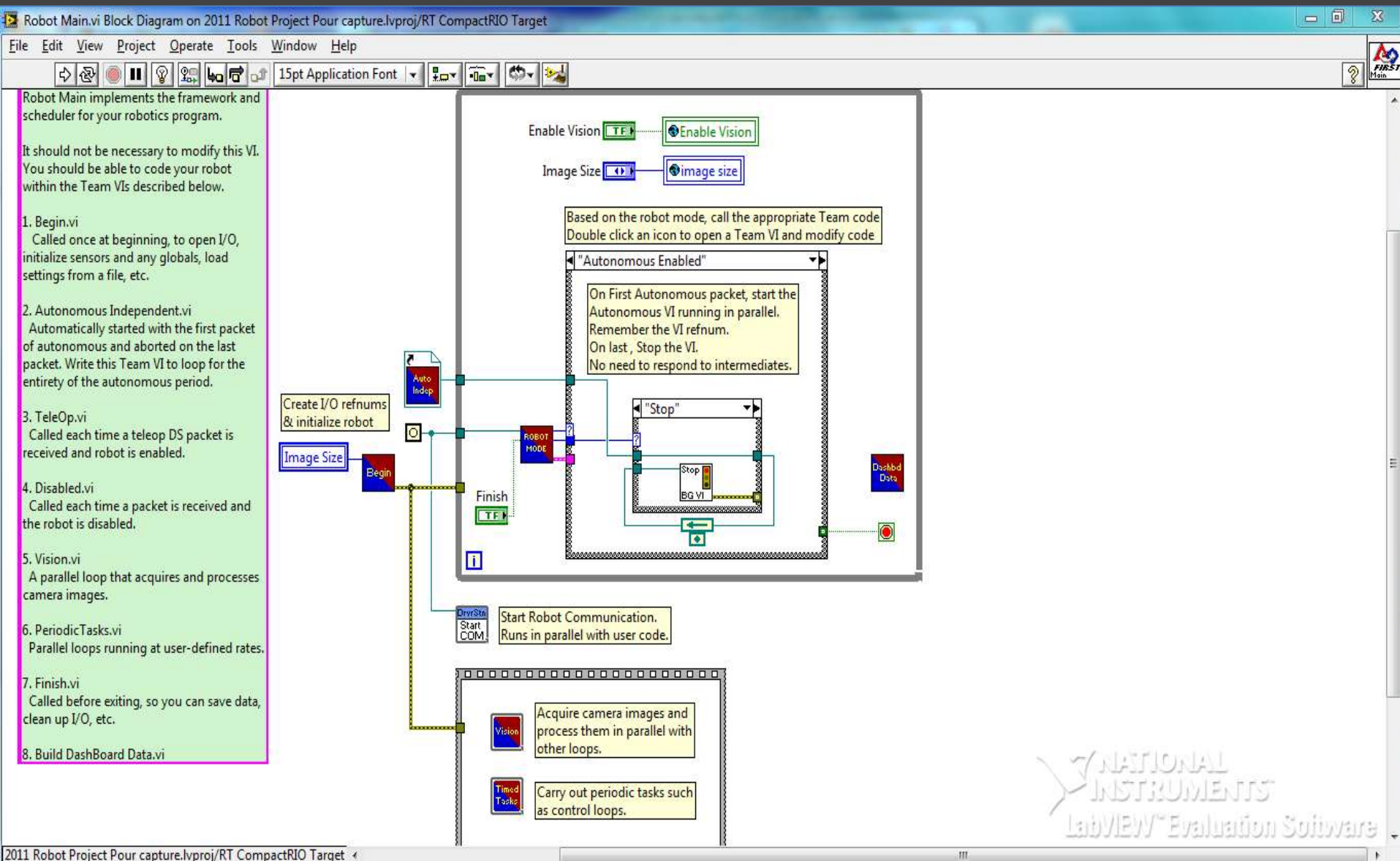
«FRC cRIO Robot Project» est un projet qui servira à la programmation du robot. En fait, le projet est composé de plusieurs VI déjà construits par FRC et nous pourrions modifier chacun de ces VI dépendant de leurs fonctions.

# Robot Main

Le VI Robot Main sert à jumeler et faire fonctionner chacun des VI ensemble. En fait, nous pouvons voir cela comme un squelette du code et chacun des autres VI est un membre du squelette. Nous n'avons pas besoin de modifier ce VI. Il faudrait même spécifier de NE PAS modifier ce VI sauf s'il est vraiment nécessaire.

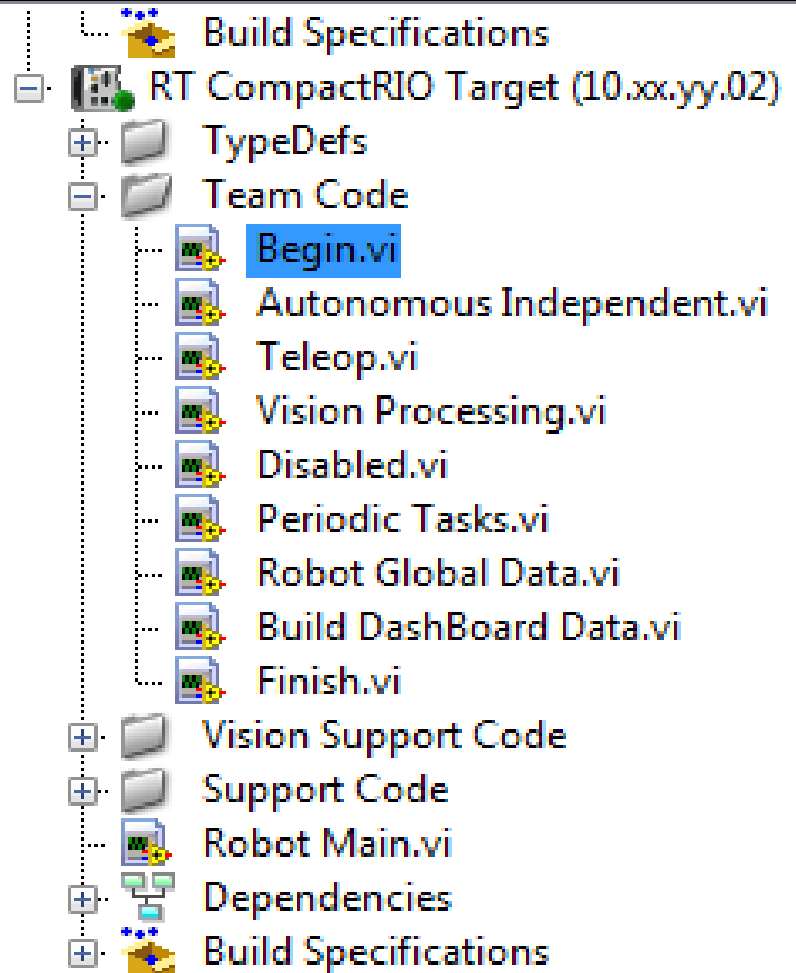


# Robot Main

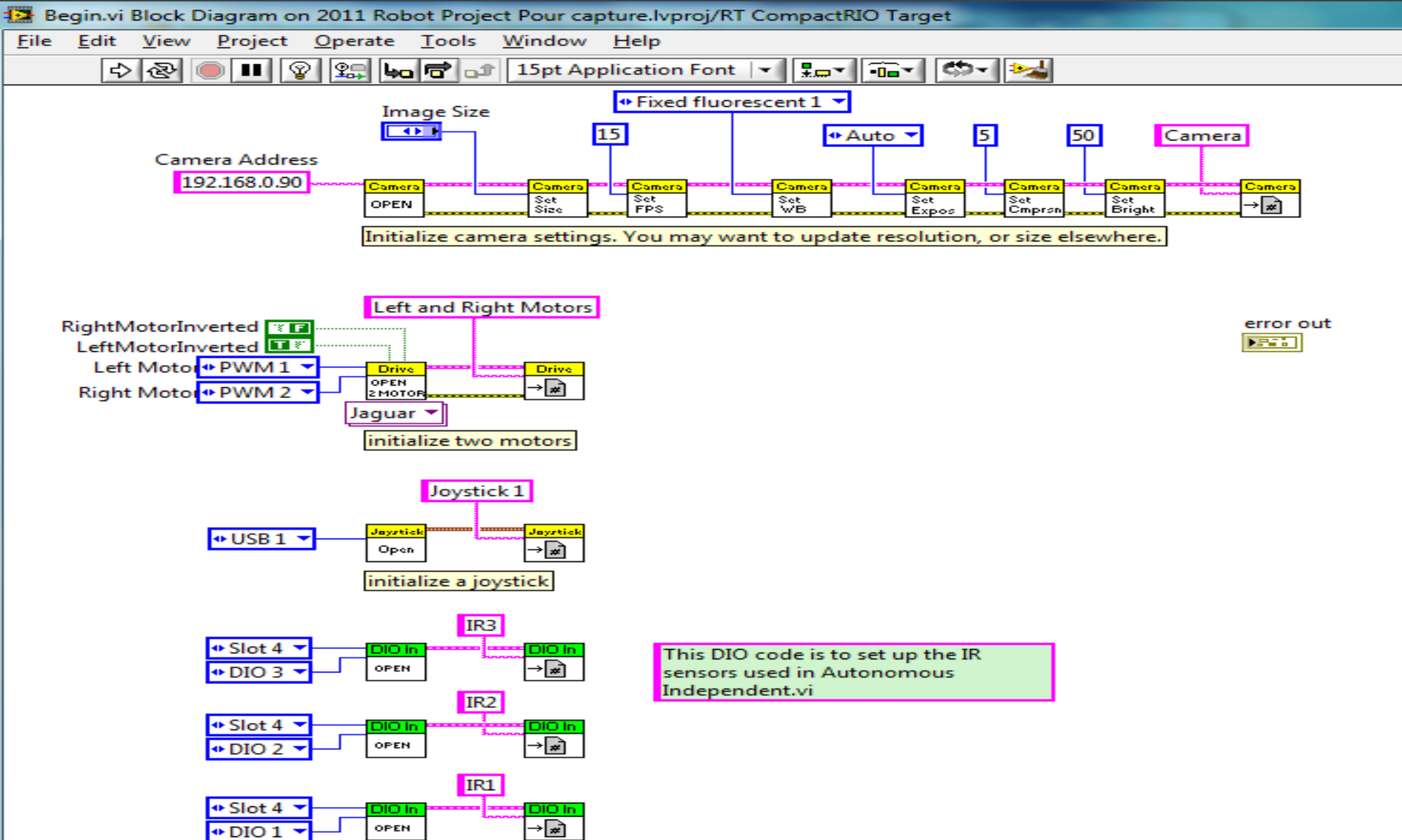


# Begin

C'est dans ce VI que nous allons initialiser les composantes telles les moteurs, les actuateurs, les senseurs, les joysticks etc. En fait, à chaque nouvelles composantes, nous devons l'initialiser dans ce VI.

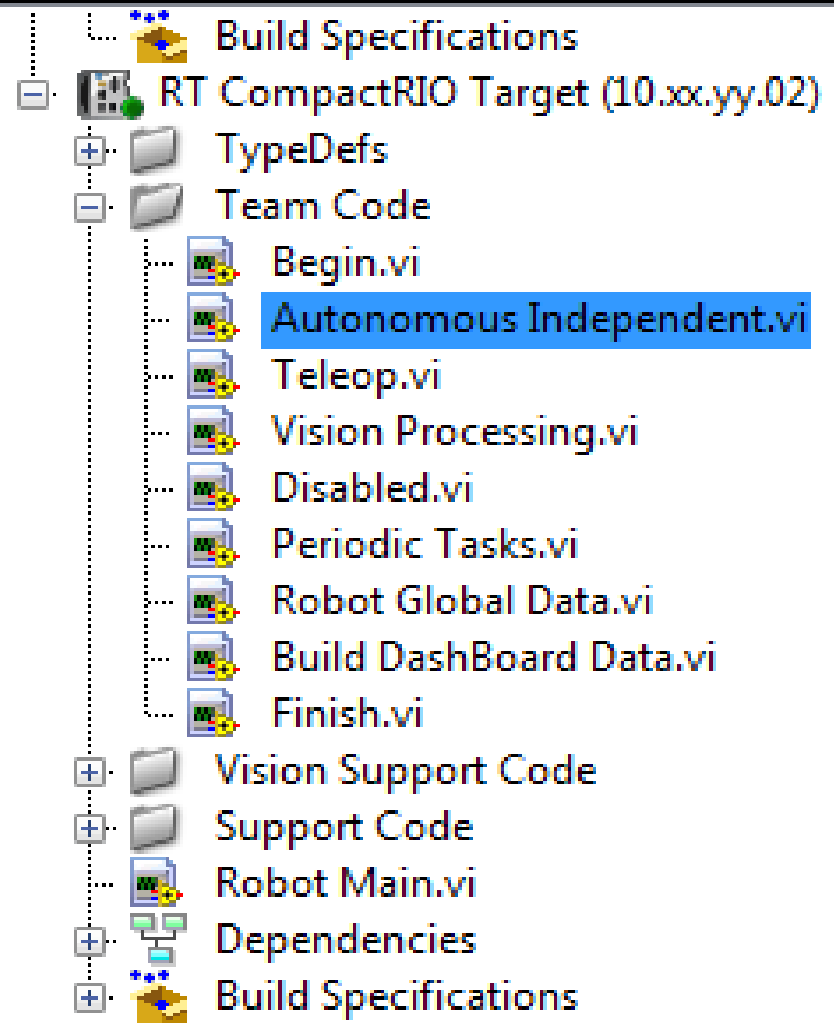


# Begin



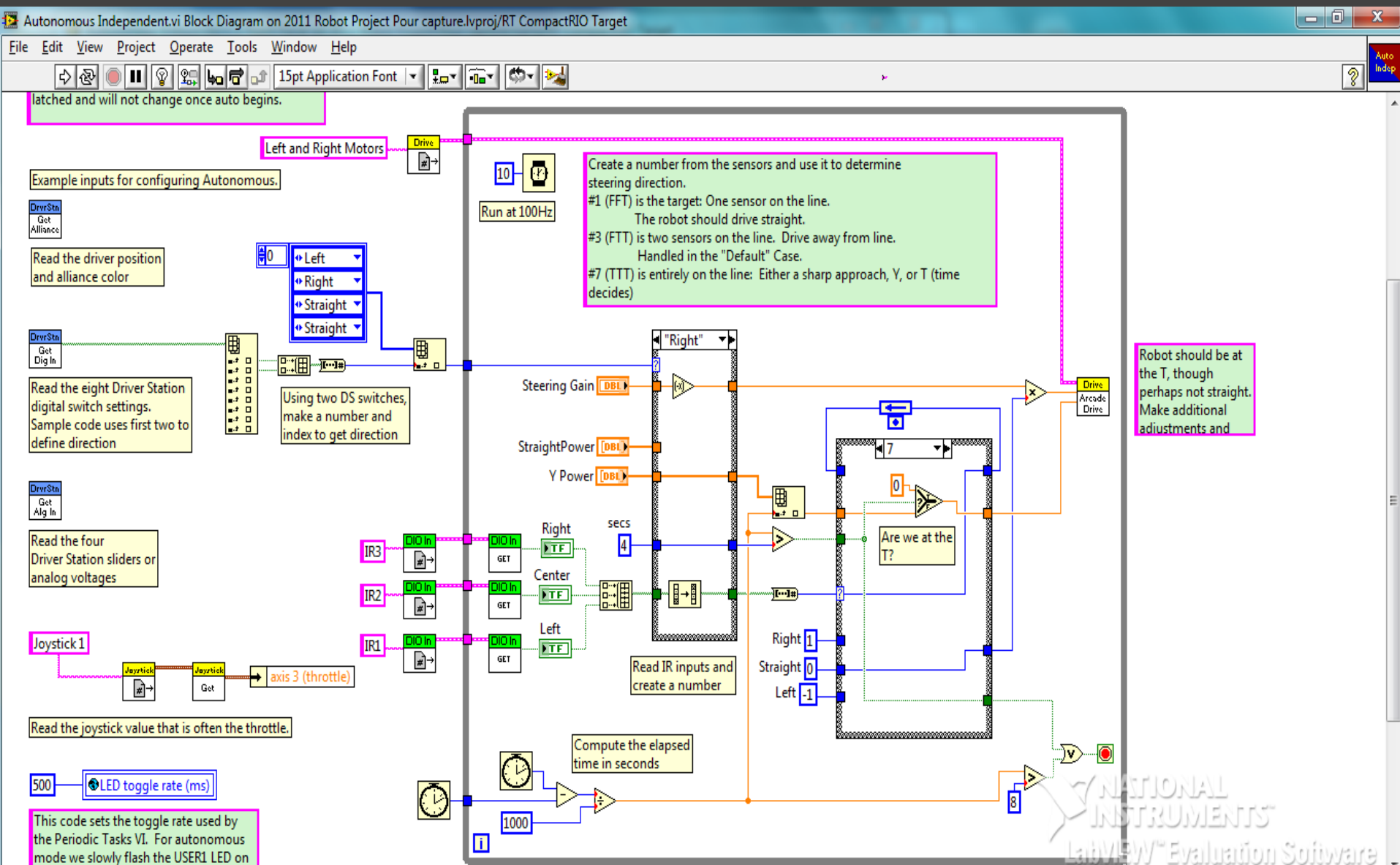
# Autonomous Independent

Dans ce VI, c'est le mode autonome qui est programmé. C'est une partie assez complexe de la programmation, mais il y a une piste de programmation, afin de nous guider vers la bonne voie.



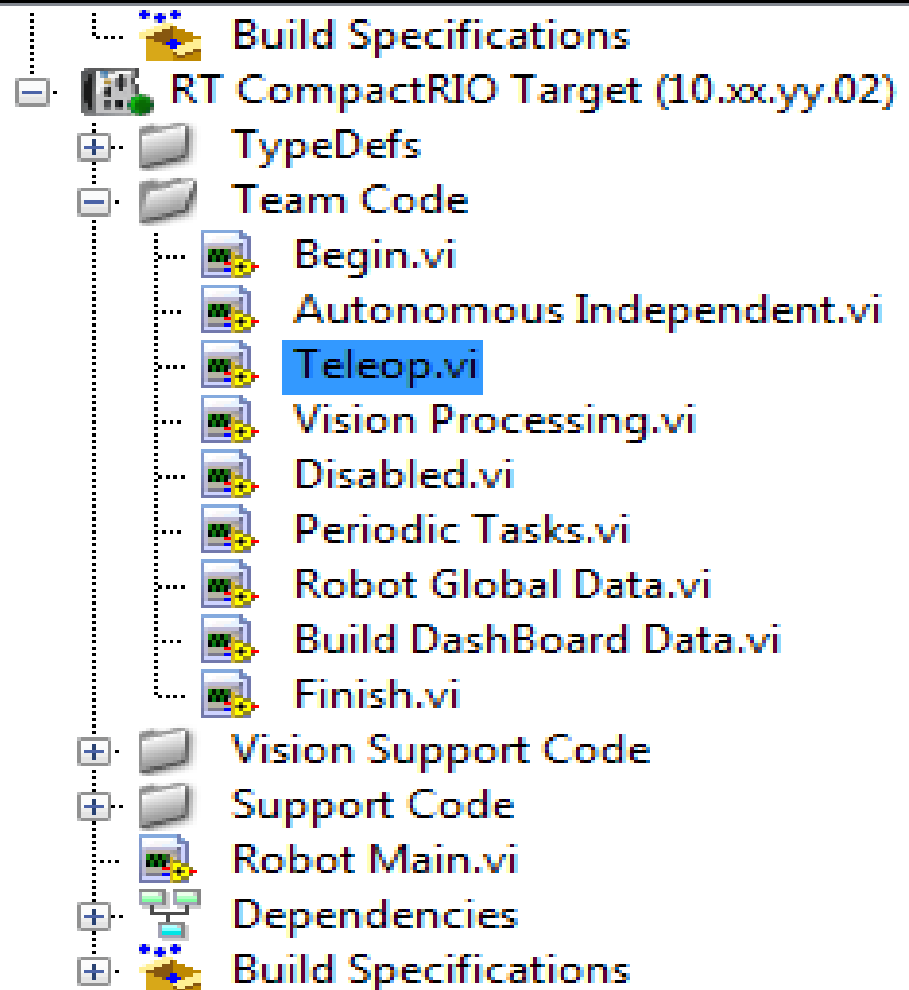


# Autonomous Independent



# Teleop

Le VI Teleop est parmi les VI les plus importants et que nous allons utiliser le plus. En fait, c'est dans ce VI que sont les moteurs, les actuateurs et les contrôles. C'est dans ce VI que l'on contrôle le robot.



# Teleop

Teleop.vi Block Diagram on 2011 Robot Project Pour capture.lvproj/RT CompactRIO Target

File Edit View Project Operate Tools Window Help



This VI is called each time a TeleOp DS packet is received. Use it to respond to new joystick or Driver Station values.

Common tasks include reading the joysticks and updating motors, and updating setpoints for periodic loops.

You can open I/O on the FIRST Call, or in the Begin.vi.

Match Info



This can help determine what has been run and for how long

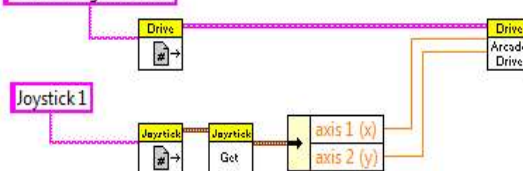
Call Context



Use to differentiate between First, Last, and Intermediate calls

Read Joystick X and Y values and update motor values

Left and Right Motors

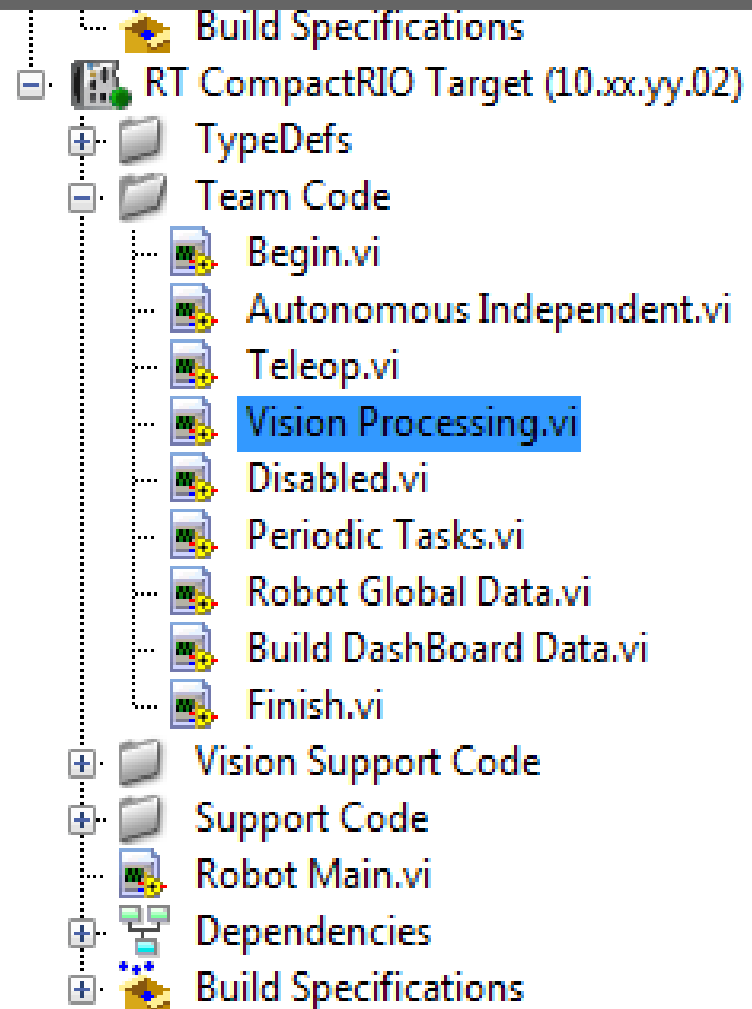


100 LED toggle rate (ms)

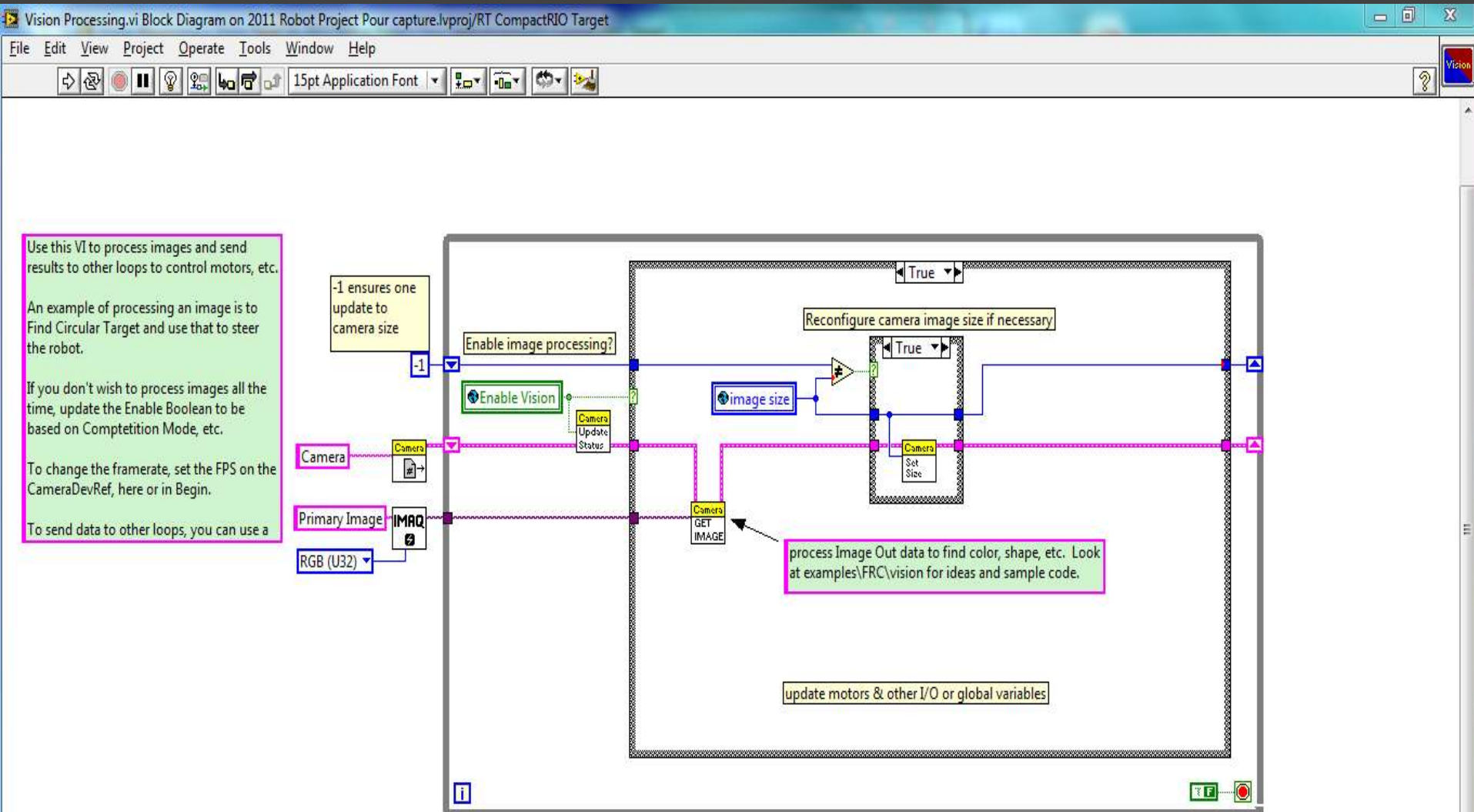
This code sets the toggle rate used by the Periodic Tasks VI. For teleop mode we flash the USER1 LED on the cRIO at a fast rate. Look at the Periodic Tasks VI to see how this global value is used.

# Vision Processing

C'est dans ce VI que la caméra entre en jeu. La programmation de la caméra et la détection rentre dans ce VI. Ici aussi, c'est une programmation assez complexe, car elle est en lien avec le VI Autonomous Independent.



# Vision Processing

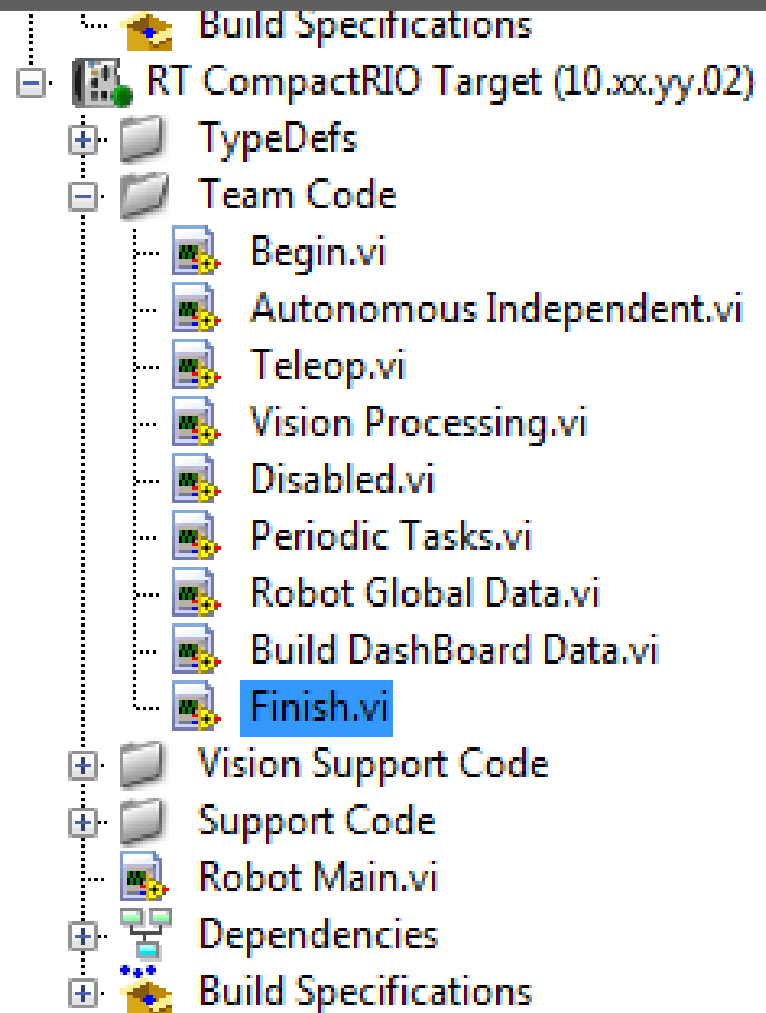


# Disabled et Periodic task

Dans ces deux VI, nous n'allons presque rien modifier. Dans le VI Disabled, on arrête les composantes s'il y a un bris ou une erreur dans le code. Et dans le VI Periodic tasks, on fait fonctionner en boucle certaines choses comme la lumière.

# Finish

Finalemment, le VI Finish est dans la même optique que le VI Begin, mais il arrête les composantes. Donc, à chaque nouvelles composantes, on ferme les nouvelles composantes.



# Finish

Finish.vi Block Diagram on 2011 Robot Project Pour capture.lvproj/RT CompactRIO Target

File Edit View Project Operate Tools Window Help



15pt Application Font

The robot controller program is about to finish.

Save collected data to files, and carry out any other tasks you'd like before exiting. You can close refsnums opened earlier in your program, but they will be cleaned up automatically anyway.

Joystick1



Left and Right Motors



Camera



Close Refs, save data, etc.

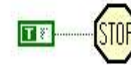
IR3



IR2



IR1



Stop all parallel loops, exit program



**Prochaine présentation :**  
**Comment initialiser les composantes**  
**dans le VI *Begin* et comment arrêter**  
**les composantes dans le VI *Finish***

